INFORMATICA, 2017, Vol. 28, No. 1, 23–44

DOI: http://dx.doi.org/10.15388/Informatica.2017.119

Developing a Two-Dimensional Categorization System for Educational Tasks in Informatics

Valentina DAGIENĖ[1]*, Sue SENTANCE[2], Gabrielė STUPURIENĖ[1]
[1]*Institute of Mathematics and Informatics, Vilnius University*
Akademijos 4, Vilnius, Lithuania
[2]*Department of Education & Professional Studies, King's College London*
150 Stamford Street, London, SE1 9NH, United Kingdom
e-mail: valentina.dagiene@mii.vu.lt, sue.sentance@kcl.ac.uk, gabriele.stupuriene@mii.vu.lt

Received: October 2016; accepted: February 2017

Abstract. Computational thinking is an increasingly important focus in computer science or informatics curricula around the world, and ways of incorporating it into the school curricula are being sought. The Bebras contest on informatics, which originated 12 years ago and now involves around 50 countries, consists of short problem-solving tasks based on topics in informatics. Bebras tasks engender the development of computational thinking skills by incorporating abstraction, algorithmic thinking, decomposition, evaluation and generalization. Bebras tasks cover a range of informatics concepts including algorithms and data structures, programming, networking, databases and social and ethical issues. Having built up a substantial number of Bebras tasks over 12 years it is important to be able to categorize them so that they can be easily accessed by the Bebras community and teachers within schools. The categorization of tasks within Bebras is important as it ensures that tasks span a wide range of topics; there have been several categorization schemes suggested to date. In this paper we present a new two-dimensional categorization system that takes account of computational thinking skills as well as content knowledge. Examples are given from recent tasks that illustrate the role that Bebras can play in the development of computational thinking skills.

Key words: Bebras contest, computational thinking, informatics concepts, informatics education, categorization, databases.

1. Introduction

Attracting youngsters to choose to study computer science or computing (widely known as informatics in Europe) at high school has always been a challenge for computer science educators. The idea of developing a contest in informatics and computer fluency for school students originated in the Institute of Mathematics and Informatics, Lithuania (Dagiene, 2005, 2006). The contest has now been extended to become a "challenge" and is being held in more than 50 countries. The challenge name "Beaver" – in Lithuanian "Bebras" – was chosen in connection with the hard-working, intelligent, goal seeking and lively wild animal.

*Corresponding author.

The Bebras challenge is an informatics education community-building model and is designed to promote informatics learning at schools by solving short concept-based tasks (Dagiene and Stupuriene, 2016). Alongside the initial goal of the Bebras project to motivate students to be more interested in topics of informatics, there is a strong intention to deepen algorithmic and operational thinking; more recently this is also extended to computational thinking.

Tasks are the most important component for developing computational thinking. In accordance with requirements, each Bebras task should include at least one informatics concept, attract children's attention by a story, picture or interactivity, be short (fit on a computer screen), and not require specific technical knowledge. Part of the task development is the categorization of tasks with the intention of having a broad range of tasks across different content areas.

In this paper we examine the relationship between computational thinking and Bebras challenges with the intention of developing a new categorization system for informatics educational tasks that includes both content areas of computer science (knowledge) and computational thinking (skills). The developed categorization system was presented and tested at an international Bebras workshop in May 2016.

2. Computational Thinking

The term computational thinking was popularized in 2006 with Jeanette Wing's article (2006) but actually originated with Seymour Papert's constructionist learning ideas (1996). There are differences between these two definitions in that Wing's definition is more focused on problem solving and Papert's definition is more focused on ideas and analysis (Mannila *et al.*, 2014). Subsequent research has expanded and interpreted the term further (Grover and Pea, 2013; Kalelioglu *et al.*, 2016; Lu and Fletcher, 2009; Selby and Woollard, 2013; Wolz *et al.*, 2011).

Computational thinking is not entirely embraced by all; critics suggest that the term is narrowing (Denning, 2009) or that computational thinking processes are widespread in other sciences (Hemmendinger, 2010). Indeed, definitions of computational thinking tend to be by example (Lee *et al.*, 2011). However, there is a huge interest in computational thinking as a means of explaining the thinking processes in computer science in school education (K-12); in USA computational thinking underlies the new curricular developments of the Computer Science Teacher Association (CSTA) and Code.org; in England, computational thinking is at the core of a mandatory new Computing curriculum from age 5–16; and Google has launched a teacher development MOOC (Massive Open Online Course) purely around computational thinking. Attention has turned to the identification of a set of skills that can be seen to comprise a broad definition of computational thinking, and that encompass the logical and problem-solving skills and thought processes that are applied by computer scientists in their work.

A broad approach to computational thinking sees it as a problem-solving process that includes (but is not limited to) the following characteristics (ISTE&CSTA, 2011):

- Formulating problems in a way that a computer can effectively carry out.
- Logically organizing and analysing data.
- Representing data through abstractions such as models and simulations.
- Automating solutions through algorithmic thinking (a series of ordered steps).
- Identifying, analysing, and implementing possible solutions with the goal of achieving the most efficient and effective combination of steps and resources.
- Generalizing and transferring this problem solving process to a wide variety of problems.

The work by Computing At School defines the five key computational thinking skills used in K-12 as 1) abstraction, 2) decomposition, 3) algorithmic thinking, 4) evaluation and 5) generalization (Csizmadia *et al.*, 2015). A pertinent question relates to how much computational thinking development is around computer programming. Lu and Fletcher (2009) take the view that computational thinking can be separated from programming, and should be taught before programming teaching starts. Definition of computational thinking includes understanding the consequences of scale, not only for reasons of efficiency but also for economic and social reasons (Wing, 2006). The Computer Science Teacher Association in USA adds broader attitudes like the ability to deal with complexity and open-ended problems, tolerance for ambiguity, and ability to work with others to achieve a common goal.

3. The Practicalities of the Bebras Challenge

The Bebras challenge exists to promote students' interest in the fundamentals of informatics from the very beginning of their school lives and to motivate students to learn and master technology (Dagiene and Futschek, 2008). One of the drivers of the Bebras community is a shared understanding that learning such fundamental concepts and principles at an early age is very important for a deeper understanding of various computer science topics. Bebras focuses on informatics concepts by supporting an understanding of computer science phenomena and the development of computational thinking.

In practical terms, the Bebras contest is held within school time, and is available to students online using contest management environments or systems set up specifically for this purpose. It is held annually in the second week of November (Bebras International Challenge on Informatics and Computational Thinking, 2016). Each contest consists of 18 to 24 questions (tasks) to be solved by the students within 45–55 minutes. The contest is designed for a range of age groups at primary and secondary school students. Different task sets are chosen for different age groups, from 6–7 years old to the oldest students in school. The participants are supervised by teachers who may choose to integrate the contest into their teaching activities. Some countries use the Bebras challenge to strengthen collaborative learning. For example, in Germany pupils solve Bebras tasks by pairs during a contest and discussions are allowed between the pairs.

In the past few years, the number of the Bebras challenge participants has been growing substantially and exceeded 1.3 million in 2015. Many countries (Lithuania and UK among

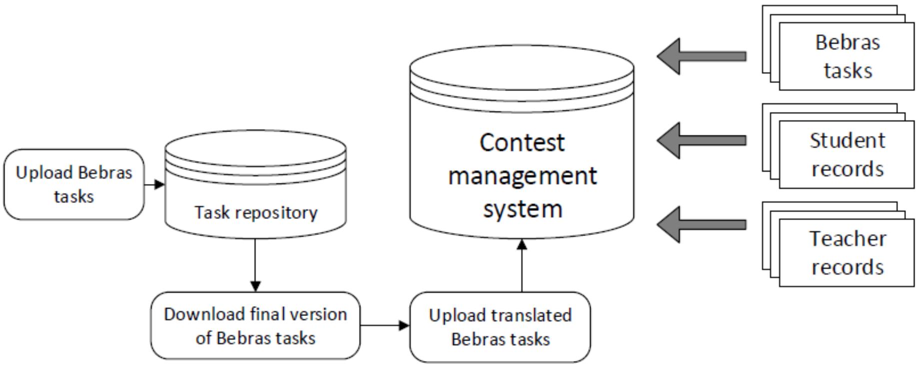


Fig. 1. Relations between task repository and CMS.

them) then organize a second round, usually face-to-face, by inviting participants who have scored highly to colleges or universities to undertake a further task solving contest.

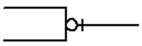
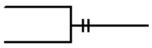
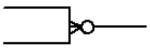
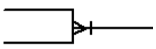
3.1. *Databases in Bebras Challenge Management Systems*

In order to run the Bebras challenge several contest (challenge) management systems (CMS) have been developed, for example, in Estonia, Finland (together with Sweden), France, Indonesia, Lithuania, Netherlands, Poland, Russia, Serbia, Slovakia, Taiwan, and Ukraine. During the week of the challenge various statistical data are collected in order to gather information about students' abilities to solve tasks and their computational thinking. In addition, this information will help to improve informatics education and improve teacher professional development.

The international community uses SVN to share effectively. The repository manages files and directories over time with files stored in a central part of the repository. The repository is much like an ordinary file server, except that it records every change ever made to user's files and directories. This allows user to recover older versions of your files and examine the history of how and when the data are changed, and who did the changes.

It can take some time (more than six months) from the initial uploading of the first version of a Bebras task into the task repository to the presentation of a final version of a task to students during the challenge (Fig. 1). The international task workshop work together each year on tasks in English; subsequently each task has to be translated into the appropriate language and then uploaded to the Bebras Contest Management System (CMS).

In Lithuania (bebras.lt), a relational database is created to store details of tasks, students and teachers. The data are stored in different tables and relations are established using primary keys or other keys known as foreign keys. MySQL is used as a database management system (DBMS).

The relational database is provided in information engineering notation (Fig. 2), where cardinalities means:  – one or zero;  – one and only one;  – zero or many;  – one or many.

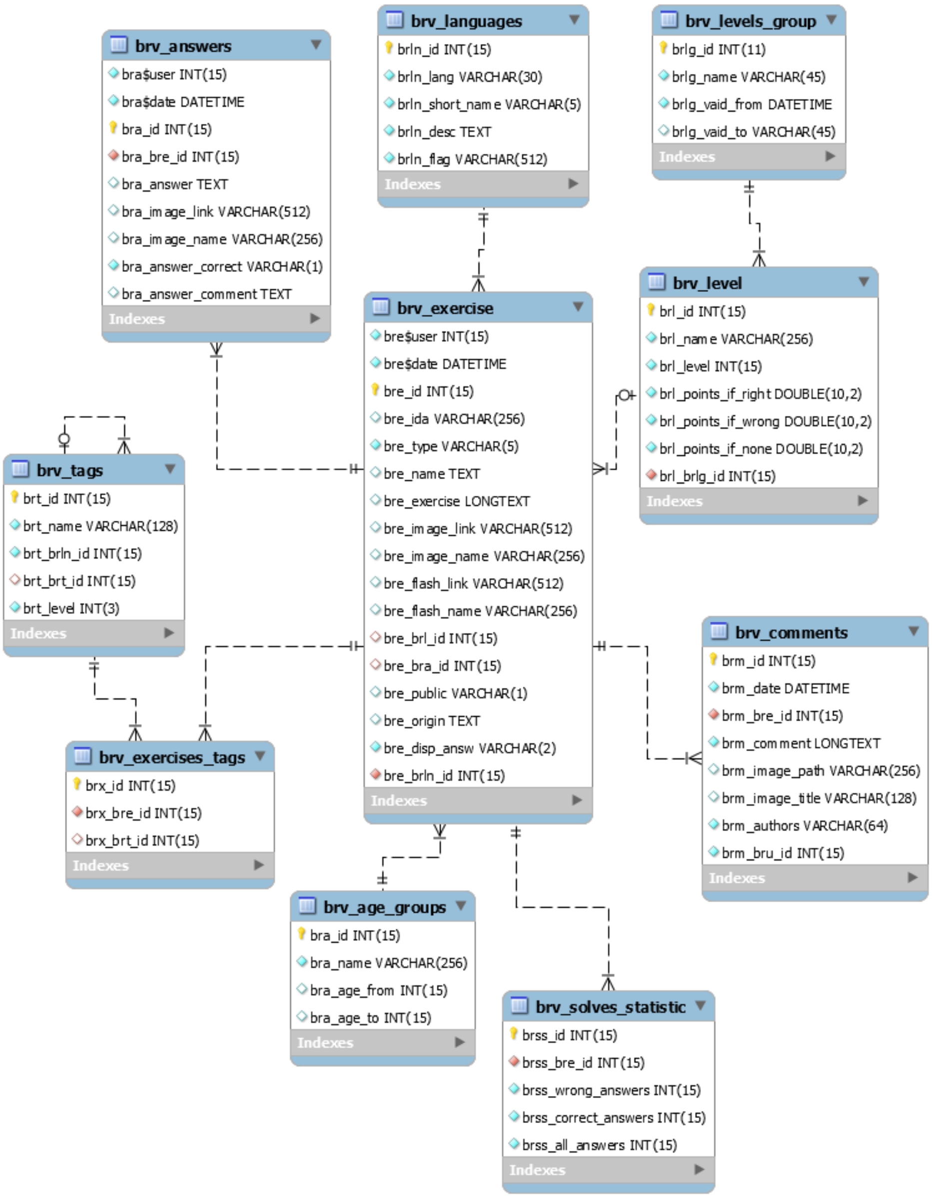


Fig. 2. The relational database underlying the Lithuanian Bebras CMS (tasks subsystem).

3.2. *Developing Tasks*

Tasks are an essential part of the challenge. Developing computer science concept-based tasks for students of different ages is rewarding but sometimes difficult for informatics educators to do. The international Bebras community comprises two computer scientists or teachers from each participating country who are responsible for the creating tasks and organization of the challenge in their countries. Naturally, each country involved has many other individuals (researchers, teachers, students) engaged in creating new tasks, running workshops to discuss informatics concepts, training teachers, and printing brochures with explanations of how the tasks can be solved. In some countries, national task-developing workshops and discussions are held and help to strengthen informatics teacher collaboration. In order to improve the quality of tasks, developers should follow these guidelines: use short sentences; repeat words or phrases; give clear definitions; show a one-to-one relationship between words and objects; use appropriate analogies; and use unambiguous wording (Pohl and Hein, 2015).

Bebras tasks can be divided in two main types: multi-choice tasks and interactive tasks. Multi-choice tasks have four non-trivial and well-defined answer choices with only one correct solution. We think of interactivity as a two-way transfer of information between a user and a machine. Thus an interactive task provides a specification of the problem and in solving it students interact directly with the computer: click spots on pictures, drag and drop objects, select list elements, etc. These tasks can be very appealing to students, especially of primary age. There is a tendency towards more interactive tasks each year. A special tool for creating interactive tasks has been developed, named the *Bebras Lodge*.

The interactivity of tasks can make the challenge more attractive where interactivity means the manipulation of objects and components. If tasks stem from real life situations, they may be engaging to older contestants. Younger students enjoy the Beaver character and find stories around characters motivating (Vaníček, 2014).

Tasks can be developed for children as young as age 6 and increasingly countries are extending the challenge to younger, primary-aged children. Contests can be a form of learning and a way to measure to some extent young children's knowledge (Tomcsányiová and Tomcsányi, 2011). However, there are design considerations around tasks for this age group. Young children may have some digital literacy skills, but it is still important to prepare the tasks for them in an age-appropriate way. Young children read more slowly and do not always understand the text correctly; children at this age need to work with concrete objects within the software and they do not understand abstraction; also children at this age may not be able to focus on a task for a long time.

The task development process is an essential part of the whole challenge. Each task must be unique and relate to at least one informatics concept. Naturally, task development can be difficult and it can be time-consuming to produce many unique tasks (Hakulinen, 2011).

4. Findings from Recent Challenges

4.1. *Participant Statistics*

In the past few years, the number of the Bebras challenge participants has been notably growing and exceeded 50 countries worldwide in 2016. The challenge is designed to promote informatics fundamentals to all students, and to be equally engaging for both boys and girls.

Table 1
Number of participants distributed by age and gender in Lithuania and UK in 2015.

Age (in years)	Grade	Number of participants (LT)	Number of participants (UK)	Number of girls (LT)	Number of girls (UK)	Number of boys (LT)	Number of boys (UK)	Unknown (UK)
6–8	1–2	0	1310	0	492	0	510	308
8–10	3–4	2374	2650	1063	949	1311	1050	651
11–12	5–6	7100	14319	3325	5543	3775	5501	3275
13–14	7–8	5810	27279	2570	10247	3240	10906	6126
15–16	9–10	6114	8119	2623	1874	3491	4530	1715
17–19	11–12	3304	2904	1031	434	2273	1865	605

Informatics (also Information Technology) is still a male-dominated discipline, but results suggest that girls aged 10–13 manage equally well (or even better) than boys in this challenge (Dagiene *et al.*, 2014). In the lower secondary school age, there are no significant differences between boys and girls in their interests and performances (Kalaš and Tomcsányiová, 2009).

One study demonstrated that pupils' performance of tasks increases with age but boys of lower grades have almost identical results to girls in the upper grades (Dagienė *et al.*, 2015); this indicates that girls and boys can be equally successful at solving such informatics tasks. However, research in Germany shows that the boys performed significantly better compared to the girls, and that pairs performed better than the singles (Hubwieser and Mühling, 2015). Another study on the performance of girls and boys in the German Bebras challenge of 2014 found that overall, the boys were more successful (Hubwieser *et al.*, 2016); in addition, the difference increased dramatically with the age of the participants. However, it was observed that girls performed better in certain types of tasks (those that were aesthetically pleasing, related to a real life situation, and were relatively easy to solve).

A moderately large number of girls take part in the challenge worldwide year by year. More than 45% of participants in Austria, France, Iceland, Macedonia, Slovakia, Sweden, Switzerland, Turkey and Ukraine in 2015 were girls. Some countries did not provide any data about gender, and 16% of Australian and 23% of UK participants were of unknown gender. France (344 976 students) and Germany (248 084) have the highest numbers of participants in the challenge; however, comparing the participant number to the population, the highest uptake comes from Slovakia (66 842 participants).

More detailed data are given in Table 1 for Lithuanian and UK participants, together with the corresponding distribution of girls and boys. The participants taking part in the challenge are divided into 6 age groups.

The distribution of participants by grades is presented in Fig. 3, respectively. The results clearly show that students grades 5 and 6 are the most active participants.

In Lithuania, 43% of students are girls and 57% boys. The number of participating girls is similar to the number of boys in grades from 3 to 6. The lowest number of participating girls is in the Seniors group (2.2 times lower than the number of boys).

Differences between boys and girls are often assumed to be a product of differences in attitudes, interests, aspirations, motivation. But in this case, the difference between the

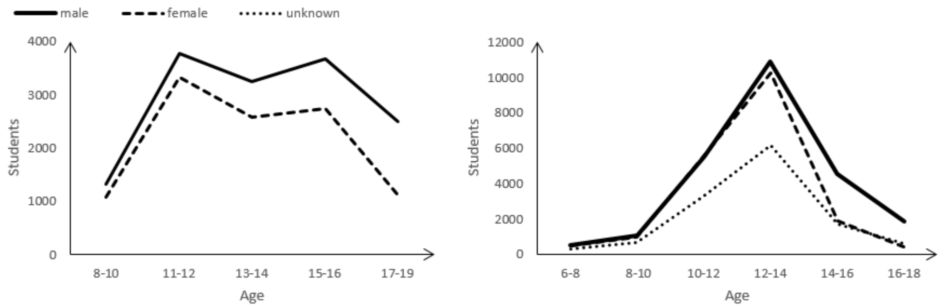


Fig. 3. Numbers of participants distributed by grades in 2015 in Lithuania and UK.

number of boys and girls is also influenced by the number of students in school at each grade. In Lithuanian schools there are more boys than girls. Also, in Lithuania children start school later, at age 7.

In terms of participation, the data show that there are similar numbers of girls and boys taking the challenge until age 12 in Lithuania and age 14 in UK. In the UK students of age 14 and above choose optional subjects to take in school. The numbers of those studying computer science will necessarily drop. As it is likely that teachers will only suggest the challenge to those students who have chosen to study computer science smaller numbers would be expected, and as we see less girls choose computer science after age 14 and particularly after age 16, this is reflected in the numbers of those taking the challenge. The number of girls in Lithuania taking the challenge in the 17–19 age group is low for the same reason. The data from UK is less conclusive as it includes a percentage of students (23%) whose gender was not declared.

In summary, the Bebras challenge can be seen to be an event that attracts girls' attention: worldwide more than 40% of participants are girls (we cannot estimate exactly because some pupils do not indicate their gender, also some countries have not yet collected statistical data). There is evidence from some countries that more girls participate at a younger age and more boys take part in the higher grades.

4.2. *Solving Bebras Tasks*

In this section we outline the achievements of students in completing the informatics tasks by presenting data from the 2015 challenge from Lithuania and the UK. The UK ran the competition in 6 age groups and Lithuania in 5.

The data shown in Fig. 4 breaks down the scores per age group to show the differences between girls and boys taking the challenge. Scores are grouped and the number of children scoring a range of marks are shown for each age group. Generally, the figures show a normal distribution of scores in solving tasks across all age groups and in both Lithuania and the UK.

In terms of achievement, the data also show a normal distribution of scores across girls and boys, meaning that girls are equally represented amongst the high scorers. The

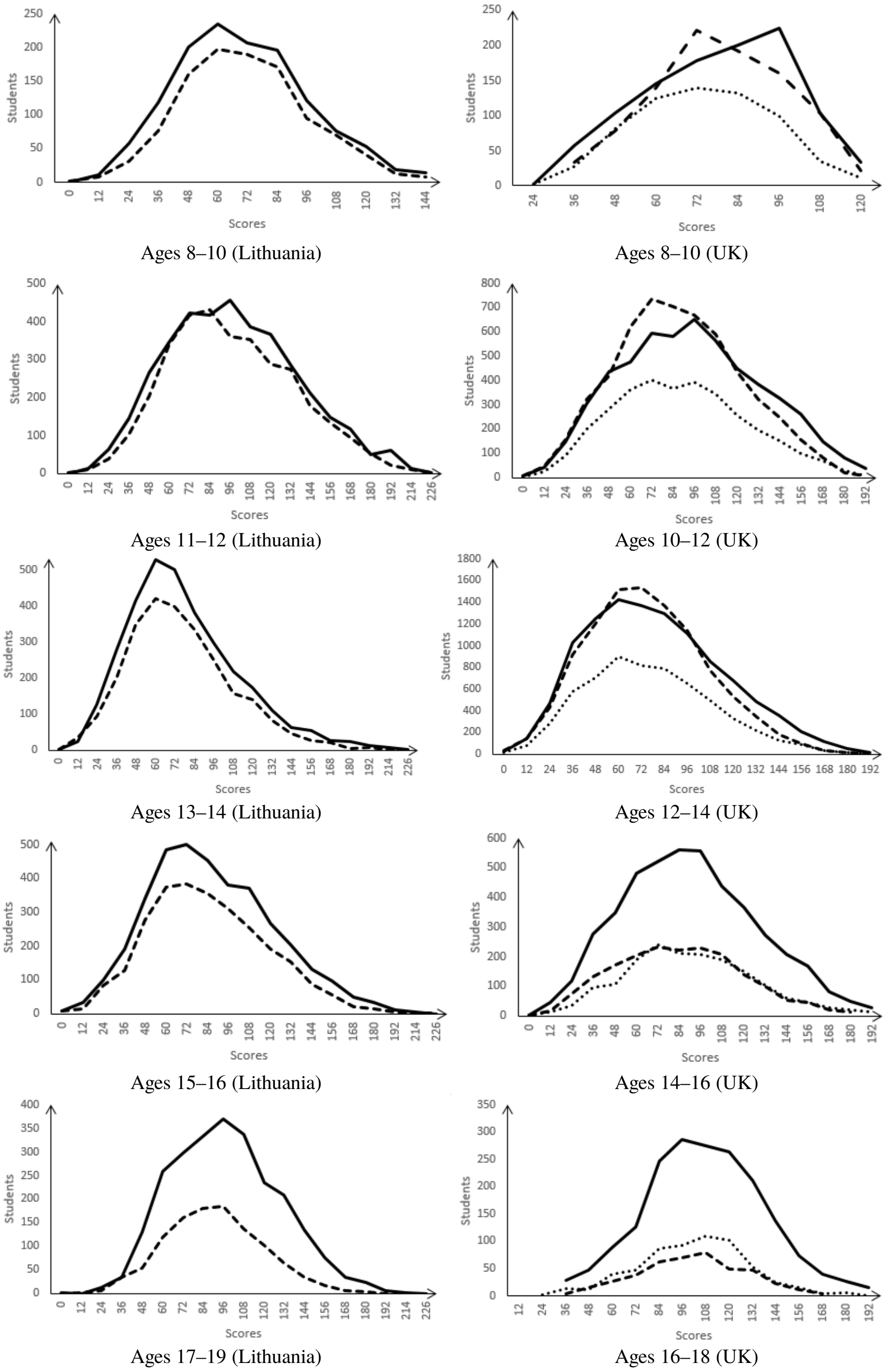


Fig. 4. Scores distributed by age groups and boys and girls in 2015 Bebras challenge in Lithuania and UK.

exception is the girls in the 10–12 age group in UK where the data suggest that the girls are scoring slightly less than the boys, in that more of them achieve the lower scores. At the senior levels, the small number of girls taking the Bebras challenge still achieve as high results as the boys in the same age group.

In our opinion, these results can provide some evidence to support the case for informatics being accessible to both genders.

In psychometrics, Item Response Theory (IRT) is a paradigm for the design, analysis, and scoring of tests, questionnaires, and similar instruments measuring abilities, attitudes, or other variables. Sets of Bebras tasks can be seen to be types of psychometric test that measure certain joint psychometric constructs (and competencies). It is assumed that the responses of students to a certain set of questions can be described by a certain psychometric model, for example by the monofactorial Rasch Model (Rost and Carstensen, 2002) with one parameter. In this case, the probability of correct answers is considered to depend on the manifestation of this construct in the following way with one parameter:

$$P(X_{ik} = 1|\theta_i, \beta_k) = \frac{\exp(\theta_i - \beta_k)}{1 + \exp(\theta_i - \beta_k)},$$

where θ_i is the parameter of person i, representing the manifestation of the psychometric construct, and β_k the parameter of item k, representing its difficulty. Under this assumption, one can estimate the person and item parameters for all k and i from the results of the contest. After this estimation, by calculating the probability P in the equation, the expected number of occurrences $E(r)$ of all possible response patterns r can be calculated. For p dichotomous questions, we have $2p$ response patterns (i.e. combinations of 0s and 1s with the length p). The expected frequencies are then compared to the actually measured pattern frequencies $O(r)$. On the differences, a X^2 statistic is applied (Hubwieser and Mühling, 2014a, 2014b, 2015; Bellettini *et al.*, 2015).

In this paper we are not focusing on evaluation so a deeper application and analysis is not provided.

5. Categorization of Bebras Tasks

In education, tasks (problems, questions) pay a significant role in engaging learners and keeping their motivation (Dagiene, 2010). Tasks should be chosen from a range of topics and contexts, using real data and an engaging scenario. The development of tasks for an educational contest is very important: they must cover fundamentals and as many subareas of discipline as possible. Moreover, the tasks have to be selected carefully, with regard to the different aspects of each task (i.e. how the topic is pitched) and interpretation of its attractiveness to pupils (whether it stimulates learning and discovery).

In general, it is important to develop a categorization system or taxonomy of educational tasks. Categorization is the basic cognitive process of arranging objects (in this case Bebras tasks) into categories, is a fundamental process in human and machine intelligence and is central to investigations and research in cognitive science (Cohen and

Lefebvre, 2005). According to Jacob (2004), categorization is the process of dividing the world into groups of entities whose members are in some way similar to each other.

Since the Bebras project began there has been an interest in classifying and categorizing tasks. Early on in the project the following seven categories were used (Opmanis *et al.*, 2006):

1) General logic;
2) ICT (Information and Communication Technology) in everyday life;
3) Practical and technical issues;
4) Information comprehension;
5) Algorithms and programming;
6) Mathematics underlying computer science;
7) History and trivia.

These categories were also used for developing new tasks as the main criteria to know which informatics topics and concepts need to be covered. Few years later the Bebras tasks' categories were revised and a modified system was proposed (Dagiene and Futschek, 2008):

1) Information comprehension;
2) Algorithmic thinking;
3) Structures, patterns and arrangements;
4) Puzzles (logical);
5) Using computer systems;
6) Social, ethical, cultural, international, and legal issues.

Countries are able to choose tasks that suit their school contexts from a large set of tasks (task pool) to which each participating country contributes every year. Lithuania uses the same task set as Austria, Germany, Hungary, Switzerland, Netherlands, as well as Finland and Sweden for all age groups except grades 3 and 4. This task set is also used by many other countries with various small changes. Although Lithuania and UK do not choose the same task set, there is a large overlap between those chosen. Here we look at the categories of tasks chosen by these two countries (the countries of the authors of this paper).

Using a categorization system (2008 version) a recent set of tasks are shown in Table 2 (only task names). This gives an indication of kinds of problems and corresponding informatics topics included in the Bebras challenge.

As can be seen from Table 2, the most popular category is *Algorithmic thinking* with 31 tasks; the category *Structures, patterns and arrangements* has less than half of this number – 16 tasks, then *Information comprehension* has got 8 tasks, *Puzzles (logical)* – 6 tasks, *Social, ethical, cultural, and legal issues* and *Using computer systems* – 3 and 2 tasks, respectively. Some tasks belong to more than one category.

This categorization system has been used to the present day; however, there have been several criticisms (for example, in the paper of Kalaš and Tomcsányiová, 2009). In addition, limitations have become apparent over time. The limitations of this categorization are as follows:

Table 2
Distribution of tasks among categories in 2015: Lithuanian and UK cases.

Task categories	Tasks identity: Lithuania & UK	Only Lithuania	Only UK
Information comprehension	Animation. Birthday balloons. Bracelet. Chakhokhbili. Fireworks. Mobiles. Quick beaver code. Stack computer		
Algorithmic thinking	Beaver logs. Biber hotel. Bowl factory. Building a chip. Chakhokhbili. Crane operating. Cross country. Drawing patterns. Drawing stars. Dream dress. Fair share. Mushrooms. Pirate hunters. Popularity. Supper power family. Theatre. Throw the dice. You won't find it	Birds Busy beaver Button game Car transportation Decorating chocolate Links um! Pencils alignment Robot the stairs	Kangaroo Reaching the target Setting the table Trains
Using computer systems	Geocaching		Setting the table
Structures, patterns and arrangements	Beaver gates. Beaver logs. Beaver lunch. Beaver the alchemist. Biber hotel. Bracelet. Drawing stars. Dream dress. Irrigation system. Popularity. Stack computer. Walnut animals. Word chains	Fried egg RAID array	Spies
Puzzles (logical)	Animal competition. Beaver gates. Geocaching. Pirate hunters	Turn the cards	Spies
Social, ethical, cultural, and legal issues	Beaver tutorials. E-mail scam	Date respect	

1) The category system is too coarse;
2) The categories overlap (e.g. *Information comprehension* and *Structures, patterns and arrangements*), which makes it difficult to assign tasks;
3) The tasks are not evenly distributed across categories;
4) Many tasks may belong to several categories;
5) The category "puzzle" is too general and of a different nature to the other categories.

Slovakian researchers had proposed an alternative categorization of the Bebras tasks (Kalaš and Tomcsányiová, 2009) using only four categories:

1) Digital literacy;
2) Programming;
3) Problem solving;
4) Data handling.

However, their system has not been adopted for use in the Bebras challenge. Their proposed categories are too general and overlap each other. Digital literacy is very broad area and can involve many skills. Problem solving is a general skill and can incorporate data handling and programming as well. This analysis informs our development of a new fine-grained classification system.

A further reason to renew the categorization system is that the Bebras project objectives have changed, reflecting the global shift of focus from computer fluency to computational thinking. Bebras is now officially named as the international challenge on informatics and computational thinking; thus the categorization of tasks must include the ways in which computational thinking skills can be developed through completing the challenge.

In order to develop a fine-grained classification, we need task categories which combine the following aspects:

1) Computational thinking skills;
2) Areas of "content competences" used to define educational standards for computer science education in school.

An example of 2) might be classifying tasks as: information and data; algorithms; languages and automata; informatics systems; and informatics, man, and society (Pohl and Westmeyer, 2015). Bebras tasks can include a wide range of concepts within informatics including algorithms and programs, both sequential and concurrent; data structures like heaps, stacks and queues; modelling of states, control flow and data flow; human-computer interaction; graphics; etc. Using a clear problem statement nearly all aspects of computer science as well as information technologies can be topics of Bebras tasks.

6. A New Categorization System: Two-Dimensional Approach

In Section 4 we saw that the categories have changed for Bebras tasks over the years the contest has been in existence. Here we introduce a revised set of categories which take into consideration the fact that completing a Bebras task demonstrates skills in computational thinking.

The area of computational thinking covers a range of different skills relating to problem-solving. The issue becomes the need to select a categorization system which is true to the definition of computational thinking whilst encompassing the range of skills that students utilize when solving Bebras tasks. There are two advantages to incorporating this into the revised category system:

1) Task development can focus more closely on how computational thinking skills are being developed or utilized;
2) Teachers and students can relate the learning from the task to their understanding of computational thinking when the tasks are discussed in the lessons following the contest.

6.1. *Informatics Concepts: Knowledge Level*

Based on previous category systems with relation to content, the content of school informatics can be divided into five areas (content categories):

1) Algorithms and programming, including logical reasoning;
2) Data, data structures and representations (includes graphs, automaton, data mining);

Table 3
Informatics content categories and keywords.

Domain	Keywords
Algorithms and programming	Algorithm; Binary search; Boolean algebra; Breadth-first search; Brute-force search; Bubble sort; Coding; Computational complexity; Constants; Constraints; Debugging; Depth-first search; Dijkstra's algorithm; Dynamic programming; Divide and conquer; Encapsulation; Function; Greedy algorithm; Heuristic; IF conditions; Inheritance; Iteration; Kruskal's algorithm; Logic gates; Loop; Maximum flow problem; Objects; Operations AND, OR, NOT; Optimization; Parameters; Prim's algorithm; Procedure; Program; Programming language; Program execution; Quick sort; Recursion; RSA algorithm; Shortest path; Searching; Sorting; Traveling salesman problem; Variables
Data, data structures and representations	Array; Attributes; Biconnected graph; Binary and hexadecimal representations; Binary tree; Character encoding; Databases; Data mining; Eulerian path; Finite-state machine; Flowcharts; Fractals; Graph; Hash table; Integer; Information; Linked list; List; Queue; Record; Stack; String
Computer processes and hardware	Cloud computing; Deadlock; Fetch-execute cycle; Grid computing; Image processing; Interpreter; Memory; Multithreading; Operating systems; Parallel processing; Peripherals; Priorities; RAID array; Registers; Scheduling; Sound processing; Translator; Turing machine
Communication and networking	Client/server; Computer networks; Cryptography; Cryptology; E-commerce; Encryption; Parity bit; Protocols; Security; Topologies
Interactions, systems and society	Classification; Computer use; Design; Ethics; Graphical user interface; Interaction; Legal issues; Robotics; Social issues, Virus

3) Computer processes and hardware (includes anything to do with how the computer works – scheduling, parallel processing);
4) Communications and networking (includes cryptography, cloud computing);
5) Interaction (Human–Computer Interaction, HCI), systems and society (all other topics!).

For practical use, when developing informatics tasks, a precise description of each category is needed. One way of achieving this uses keywords. A suggested set of keywords for each of these informatics domain areas is shown in Table 3.

Keywords are important to assist in the categorization. They will also be important to teachers who wish to find tasks that fit with the topic being taught in the curriculum (Dagiene and Sentance, 2016; Yang and Park, 2014). Therefore, keywords information should be retained with the task to help Bebras users select from previous tasks and identify teaching topics around Bebras tasks.

6.2. *Computational Thinking: Skills Level*

A suggested categorization of computational thinking skills follows the work of Selby and Woollard (2013) and which has been adopted by Computing At School in the UK in developing guidance on computational thinking for teachers (Csizmadia *et al.*, 2015). This describes aspects of computational thinking skills exhibited by learners as falling into the five categories below:

Table 4
Computational thinking skills and ways to identify them.

Computational thinking skill	How to spot use of that skill
Abstraction	Removing unnecessary details; Spotting key elements in problem; Choosing a representation of a system
Algorithmic thinking	Thinking in terms of sequences and rules; Executing an algorithm; Creating an algorithm
Decomposition	Breaking down tasks; Thinking about problems in terms of component parts; Making decisions about dividing into sub-tasks with integration in mind, e.g. deduction
Evaluation	Finding best solution; Making decisions about good use of resources; Fitness for purpose
Generalization	Identifying patterns as well as similarities and connections; Solving new problems based on already-solved problems; Utilizing the general solution, e.g. induction

1) Abstraction;
2) Algorithmic thinking;
3) Decomposition;
4) Evaluation;
5) Generalization.

The use of keywords will be slightly different for computational thinking skills. Classifiers need to know how to identify if that skill may be used to solve that task (Table 4). One of the difficulties is that we can only presume how the learner solves the task which may be a different way to the way the task setter might solve the task. This means that more than one computational thinking skill may be associated with each task. We are suggesting a maximum of three.

6.3. *A Two-Dimensional Categorization System*

Incorporating both described categorization systems (Sections 5.1 and 5.2), we can compose a two-dimensional system which can be represented as shown in Table 5. The suggested categorization system incorporates both computational thinking skills and informatics concepts in the classification of tasks.

The presentation of this schema as a 2-D matrix merely indicates that every computational thinking skill can occur with each of the concept ideas – there is no dependency between the two classifiers. In practical terms, a task should be allocated to one informatics content area only but may have up to three computational thinking skills identified. Computational thinking skills are more difficult to clearly define and identify in a task as they are dependent on the approach taken to solve the problem; thus some flexibility is needed here.

Table 5
Two-dimensional categorization system.

	Algorithms and programming	Data, data structures and representations	Computer processes and hardware	Communication and networking	Interactions, systems and society
Abstraction					
Algorithmic thinking					
Decomposition					
Evaluation					
Generalization					

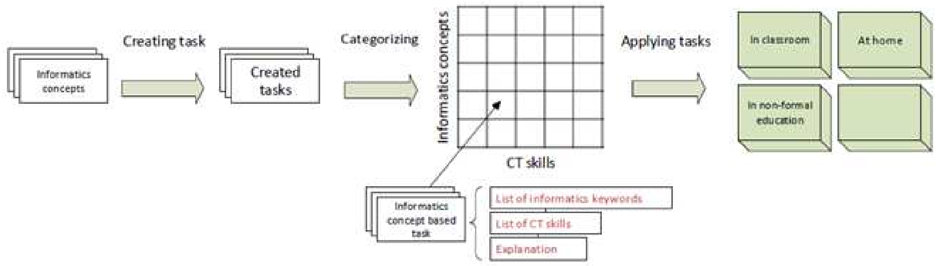


Fig. 5. Tasks creating, categorizing and using process.

Table 6
A template table for task categorization using the two-dimensional categorization system.

Name of task	Informatics domain	Keywords (⩽ 3)	CT Skill (⩽ 3)

The categorization system could be used in addition to encourage the development of tasks that use a variety of computer science topic areas as well as computational thinking skills. On the other hand, this system helps teachers of informatics to choose the content of lesson and helps effectively to select the tasks according to the particular topic (Fig. 5).

The matrix presented in Table 5 demonstrates that this schema can be seen as two-dimensional. In practical terms, a template has been designed for developers to assign categories to tasks, including keywords (Table 6).

Although more complex, this new system captures more information about the task in a way that will be accessible and will support both task setter and teacher.

7. Evaluation of the New Categorization System

Before arriving at the system described above, other suggestions have been considered along the way. In seeking to both address the limitations of previous categorization and to have a system that was easily understood the authors needed to consult with informatics educators, particularly with the Bebras community.

An expert evaluation was planned during the annual Bebras community meeting in May 2016, where there would be representatives from almost every country in the community. There were around 80 members of the Bebras community present and an early version of the proposed system was explained and exemplified.

Feedback was taken from the members of the Bebras community and members were given tasks to categorize according to a version of this system. Feedback was taken both verbally and in writing and the comments of the community were used to refine the system.

In particular, the experience of sharing the categorization illuminated some of the points raised above:

- Clear illustration of computational thinking skills with examples is needed as we cannot have assumed that any knowledge of these is shared in the community;
- Keywords are essential both to illustrate the informatics domains and the computational thinking skills to assist categorizers;
- Categorizers need to focus on the experience of the student solving the problem and not the task setter (expert) in assigning both concept and Computational Thinking skill.

The informal evaluation of the categorization informed the proposed schema, but it will be formally evaluated in the next round of the Bebras challenge. In this paper we are seeking to propose this as a new categorization and exemplify it with a number of recent tasks.

We present here some typical Bebras tasks and discuss the concepts that may be learned by these tasks. These tasks illustrate that we want to let the students discover the informatics concepts by themselves. To solve the tasks, the participants have to explore the stated problem domain and have to work with data, structures, activities and problems which are typical for informatics. Usually there are many different ways for finding a solution and thus the contest also supports students that prefer different types of solution strategies.

Six examples follow which illustrate the way that individual tasks can be categorized (Table 7). The tasks have been written by representatives from various countries within the Bebras community. Each task has been further developed within the community at task workshops.

8. Discussion

As the Bebras challenge has become so widely known and used in schools in so many countries, any change of categorization system should be evaluated carefully. In our initial development of the new categories we have consulted with members of the Bebras community and presented an earlier version of the proposed system. The resultant system incorporates feedback from Bebras task developers and organizers from several countries. There are additional considerations that will have an impact on the implementation.

Firstly, this approach is more complex than the previous one. This will give us a more finely-grained classification that will produce much more useful output as the number of available tasks for teaching increases. However, a more finely-grained system requires more knowledge and understanding to implement correctly. It may be that task developers

Table 7
Short presentation of selected task examples.

Name of task	Short description of task	Informatics domain	Keywords (⩽ 3)	CT Skill (⩽ 3)
Bowl factory (Malaysia)	A sets of 6 different sizes bowls are moving on a long conveyor belt. The bowls need to be sorted by swapping any two neighbouring bowls. Question is how many workers should be put along the conveyor to sort the set of bowls	Algorithms and programming	An algorithm Sorting Bubble sort algorithm	Algorithmic thinking
Beaver the alchemist (Russia)	There is a schema which shows how objects can be converted to another objects. The given rules explained how many objects we need in order to get a particular object. The task can be solved by interpreting rules of converting	Data, data structures and representations	Graph Schema Data structures	Algorithmic thinking Decomposition
Reaching the target (Belgium)	In this task an archer needs to locate a target with his arrow using the minimum number of attempts	Algorithms and programming	Bisection method O(log n) complexity Binary search	Algorithmic thinking Abstraction
You won't find it (Belgium)	This task involves decoding a message by carrying out a sequence of steps	Communications and networking	Cryptography Algorithm Ciphering	Algorithmic thinking
Chakhokhbili (Russia)	In this task, a series of cooking steps are given and the task is to find the shortest time that the whole meal can be cooked using several pans in parallel	Computers processes and hardware	Parallel processing Scheduling Optimization	Decomposition Evaluation
Beaver tutorials (Hungary)	This task is about the use of copyright legislation and procedures to decide what is an ethical approach in a given situation	Interactions, systems and society	Copyright Ethics	Evaluation

in different countries are not able (or willing) to assign this level of detailed categorization to each Bebras task. The question will arise as to who will carry out the categorization after the tasks have been chosen by the Bebras international workshop participants.

Secondly, computational thinking may not be familiar to all participating countries and understanding of the component skills presented here may not be shared. The community will need clear examples of computational thinking skills in Bebras tasks and explanations should be available to ensure some consistency of allocation of computational thinking skills to task.

Thirdly, related to this, we will need to develop some precision in allocating computational thinking skills to tasks. The description by Wing (2008) may lead us to think that computational thinking is everywhere and the composite skills all crop up in all tasks. A liberal interpretation such as this may render the computational thinking skill allocation to be meaningless. Computational thinking skills should only be allocated to a task where there is some element of computer science in the task that develops this skill.

With due attention to the points raised above, the purpose of this development is to build up a bank of Bebras tasks from recent years which are categorized using this framework. This will enable teachers and others to find useful tasks that they can use in the curriculum. It will also help task developers to focus on writing tasks around topics that are under-represented in the bank of tasks. An online search facility could be implemented to assist teachers looking for Bebras tasks on certain topics via keywords, concepts or computational thinking skills.

9. Conclusion

In this paper a categorization for Bebras tasks has been presented. This will be useful to task developers who hopefully will be able to spread new tasks over a range of categories. It will be most useful to teachers in school once tasks are classified according to this system: tasks can then be selected from previous contests that fit particular criteria to support either curriculum teaching or practice for the Bebras contest.

Tasks are very important both for competitors (students) and task developers (teachers): students should be encouraged to think about computer science, educators should think about harmonization of syllabus of informatics. Creative, interesting tasks are the main driver for the Bebras contests and motivation students to learn informatics.

Acknowledgements. The authors thank all members of the international Bebras challenge community that took part in task development and in this way influenced the outcome of this paper. In addition, we are grateful to Eljakim Schijvers (Netherlands) and Rimantas Žakauskas (Lithuania), CMS developers, for providing Bebras statistical data.

References

Bebras International Challenge on Informatics and Computational Thinking (2016). Available via internet: http://www.bebras.org/en/facts. Accessed 20 September 2016.

Bellettini, C., Lonati, V., Malchiodi, D., Monga, M., Morpurgo, A., Torelli, M. (2015). How challenging are Bebras tasks? An IRT analysis based on the performance of Italian students. In: *Proceedings of the 2015 ACM Conference on Innovation and Technology in Computer Science Education*. ACM, pp. 27–32.

Cohen, H., Lefebvre, C. (Eds.) (2005). *Handbook of Categorization in Cognitive Science*. Elsevier.

Csizmadia, A., Curzon, P., Dorling, M., Humphreys, S., Ng, T., Selby, C., Woollard, J. (2015). Computational thinking: a guide for teachers. Available via internet: http://computingatschool.org.uk/computationalthinking. Accessed 20 September 2016.

Dagiene, V. (2005). Competition in information technology: an informal learning. In: *EuroLogo 2005: The 10th European Logo Conference, Digital Tools for Lifelong Learning, Warsaw, Poland, 28–31 August*, pp. 228–234.

Dagiene, V. (2006). Information technology contests – introduction to computer science in an attractive way. *Informatics in Education*, 5(1), 37–46.

Dagiene, V. (2010). Sustaining informatics education by contests. In: Hromkovic, J., Kralovic, R., Vahrenhold, J. (Eds.), *Lecture Notes in Computer Science*, Vol. 5941, pp. 1–12.

Dagiene, V., Futschek, G. (2008). Bebras international contest on informatics and computer literacy: criteria for good tasks. In: Mittermeir, R.T., Syslo, M.M. (Eds.), *Informatics Education – Supporting Computational Thinking. Lecture Notes in Computer Science*, Vol. 5090. Springer, pp. 19–30.

Dagiene, V., Sentance, S. (2016). It's computational thinking! Bebras tasks in the curriculum. In: *Lecture Notes in Computer Science*, Vol. 9973, pp. 28-39.

Dagiene, V., Stupuriene, G. (2016). Bebras – a sustainable community building model for the concept based learning of informatics and computational thinking. *Informatics in Education*, 15(1), 25–44.

Dagiene, V., Mannila, L., Poranen, T., Rolandsson, L., Stupuriene, G. (2014). Reasoning on children's cognitive skills in an informatics contest: findings and discoveries from Finland, Lithuania, and Sweden. In: *Lecture Notes in Computer Science*, Vol. 8730, pp. 66–77.

Dagienė, V., Pėlikis, E., Stupurienė, G. (2015). Introducing computational thinking through a contest on informatics: problem-solving and gender issues. *Informacijos mokslai*, 73, 43–51.

Denning, P.J. (2009). Beyond computational thinking. *Communications of the ACM*, 52(6), 28–30.

Grover, S., Pea, R. (2013). Using a discourse-intensive pedagogy and Android's App inventor for introducing computational concepts to middle school students. In: *Proceedings of the 44th SIGCSE Technical Symposium on Computer Science Education*. ACM.

Hakulinen, L. (2011). Survey on informatics competitions: developing tasks. *Olympiads in Informatics*, 5, 12–25.

Hemmendinger, D. (2010). A plea for modesty. *ACM Inroads*, 1(2), 4–7.

Hubwieser P., Mühling, A. (2014a). Competency mining in large data sets – preparing large scale investigations in computer science education. In: *Proceedings of the International Conference on Knowledge Discovery and Information Retrieval*, pp. 315–322.

Hubwieser, P., Mühling, A. (2014b). Playing PISA with Bebras. In: *Proceedings of the 9th Workshop in Primary and Secondary Computing Education*. ACM, pp. 128–129.

Hubwieser, P., Mühling, A. (2015). Investigating the psychometric structure of Bebras contest: towards measuring computational thinking skills. In: *Learning and Teaching in Computing and Engineering (LaTiCE)*. IEEE, pp. 62–69.

Hubwieser, P., Hubwieser, E., Graswald, D. (2016). How to attract the girls: gender-specific performance and motivation in the Bebras challenge. In: *Lecture Notes in Computer Science*, Vol. 9973, pp. 40–52.

ISTE&CSTA (International Society for Technology in Education & the Computer Science Teachers Association) (2011). Operational definition of computational thinking for K-12 education. Available via internet: https://csta.acm.org/Curriculum/sub/CurrFiles/CompThinkingFlyer.pdf. Accessed 20 September 2016.

Jacob, E.K. (2004). Classification and categorization: a difference that makes a difference. *Library Trends*, 52(3).

Kalaš, I., Tomcsányiová, M. (2009). Students' attitude to programming in modern informatics. *Informįtica na Educa-ção: Teoria & Prática, Porto Alegre*, 12(1), 127–135.

Kalelioglu, K. Gülbahar, Y., Kukul, V. (2016). A framework for computational thinking based on a systematic research review. *Baltic Journal of Modern Computing*, 4(3), 583–596.

Lee, I., Martin, F., Denner, J., Coulter, B., Allan, W., Erickson, J., Malyn-Smith, J., Werner, L. (2011). Computational thinking for youth in practice. *ACM Inroads*, 2(1), 32–37.

Lu, J.J., Fletcher, G.H. (2009). Thinking about computational thinking. *ACM SIGCSE Bulletin*, 41(1), 260–264.

Mannila, L., Dagiene, V., Demo, B., Grgurina, N., Mirolo, C., Rolandsson, L., Settle, A. (2014). Computational thinking in K-9 education. In: *Proceedings of the Working Group Reports of the 2014 on Innovation & Technology in Computer Science Education Conference*, *ITiCSE-WGR 2014*. ACM, New York, pp. 1–29.

Opmanis, M., Dagiene, V., Truu, A. (2006). Task types at "Beaver" contests. *Information Technologies at School*, 509–519.

Papert, S. (1996). An exploration in the space of mathematics educations. *International Journal of Computers for Mathematical Learning,* 95–123.

Pohl, W., Hein, H.W. (2015). Aspects of quality in the presentation of informatics challenge tasks. In: *Lecture Notes in Computer Science*, Vol. 9378, pp. 21–32.

Pohl, W., Westmeyer, J. (2015). Content categories for informatics tasks. In: *The Proceedings of International Conference on Informatics in Schools: Situation, Evolution and Perspectives, ISSEP 2015*, p. 61.

Rost, J., Carstensen, C.H. (2002). Multidimensional rasch measurement via item component models and faceted designs. *Applied Psychological Measurement*, 26(1), 42–56.

Selby, C., Woollard, J. (2013). Computational thinking: the developing definition. Available via internet: http://eprints.soton.ac.uk/356481.

Tomcsányiová, M., Tomcsányi, P. (2011). Little beaver – a new bebras contest category for children aged 8–9. In: *Informatics in Schools. Contributing to 21st Century Education*. Springer, Berlin, pp. 201–212.

Vaníček, J. (2014). Bebras informatics contest: criteria for good tasks revised. In: *Lecture Notes in Computer Science*, Vol. 8730, pp. 17–28.

Wing, J.M. (2006). Computational thinking. *Communications of the ACM*, 49(3), 33–35.
Wing, J.M. (2008). Computational thinking and thinking about computing. *Philosophical Transactions of the Royal Society*, 366, 3717–3725.
Wolz, U., Stone, M., Pearson, K., Pulimood, S.M., Switzer, M. (2011). Computational thinking and expository writing in the middle school. *ACM Transactions on Computing Education (TOCE)*, 11(2), 9.
Yang, S., Park, S. (2014). Teaching some informatics concepts using formal system. *Informatics in Education*, 13(2), 323.

V. Dagienė is a professor and principal researcher at Vilnius University Institute of Mathematics and Informatics. She has published over 200 scientific papers and more than 50 textbooks in informatics for high schools. She has been working in various expert groups and work groups, organizing the olympiads in informatics among students, also engaged in localization of software and educational programs, e-learning, and problem solving. She is an Executive Editor of international journals *Informatics in Education* and *Olympiads in Informatics*. She has participated in several EU-funded R&D projects, as well as in a number of national research studies connected with technology and education.

S. Sentance is a senior lecturer in computer science education at King's College London in the UK. She has considerable experience in training pre-service and in-service teachers of computer science. She has authored a range of papers on teacher professional development and curriculum change in Computing and works with the Bebras team in the UK.

G. Stupurienė is a doctoral student at Vilnius University Institute of Mathematics and Informatics at the Department of Informatics Methodology. She has been working with Bebras challenge since 2010. As a Master student she worked on Conceptualization of Informatics Fundamentals through Bebras Tasks of earlier years. Her main research focus is developing informatics concepts based educational model for schools.